Defense Technical Information Center Compilation Part Notice

ADP014108

TITLE: Using RANS Mean Flow Fields in Numerical Aeroacoustics Simulations [CAA]

DISTRIBUTION: Approved for public release, distribution unlimited
Availability: Hard copy only.

This paper is part of the following report:

TITLE: Aging Mechanisms and Control. Symposium Part A - Developments in Computational Aero- and Hydro-Acoustics. Symposium Part B - Monitoring and Management of Gas Turbine Fleets for Extended Life and Reduced Costs [Les mecanismes vieillissants et le controle] [Symposium Partie A - Developpements dans le domaine de l'aeroacoustique et I'hydroacoustique numeriques] [Symposium Partie B ...

To order the complete compilation report, use: ADA415749

The following component part numbers comprise the compilation report:
ADP014092 thru ADP014141

Using RANS Mean Flow Fields in Numerical Aeroacoustics Simulations (CAA)

M. Lummer, H.A. Grogger, J.W. Delfs
Deutsches Zentrum für Luft- u. Raumfahrt (DLR)
Institut für Aerodynamik und Strömungstechnik
Postfach 3267
D-38022 Braunschweig
Germany

Abstract

One way to perform CAA simulations is to split the flow field in a steady mean flow and turbulent and acoustic fluctuations. The small acoustic fluctuations can then be calculated from linearized Euler equations (LEE). The mean flow can be supplied by the solution of the Reynolds averaged Navier-Stokes (RANS) equations. Since different numerical requirements exist for the numerical grids of RANS and LEE simulations, an interpolation procedure between RANS and LEE grids is necessary. The following paper presents an interpolation procedure between different structured multi-block grids and its application to RANS/CAA simulations of a generic wing section.

1. Introduction

The most accurate way to calculate the generation and propagation of sound in flows is the solution of the unsteady conservation laws of mass, momentum, and energy. Since this direct numerical simulation is even for simple flows an extremely expensive task, a lot of approximative methods have been proposed in order to attack real world problems. One of them is the splitting of the flow field in mean flow and turbulent and acoustic fluctuations. The mean flow is approximated by a solution of the Reynolds averaged equations (RANS), the turbulent fluctuations e.g. by an SNGR model (Stochastic Noise Generation and Radiation) [2,3,4,7], and the acoustic fluctuations by the solution of the linearized Euler equations (LEE). This approach is utilized in the german SWING project (Simulation of Wing-flow Noise Generation), which shall provide a general computer code for the calculation of airframe noise, especially that generated by high lift devices.

Using this RANS-LEE approach, it is generally necessary to use different grids for both calculations. The reason for this are the different numerical requirements in both cases. E.g. the RANS calculations need very fine grids in the boundary layers of the body, whereas the LEE require a sufficient fine resolution in the farfield of the body. Therefore the RANS solution has to be provided on the CAA grid by means of interpolation. The present paper discusses this transfer of RANS data into the CAA perturbation code in view of robustness, accuracy as well as the numerical implications on CAA simulations of airframe noise.

This article is structured as follows. First a brief overview of RANS calculations of a generic wing section will be given. Then the CAA grid and the interpolation procedure between the RANS grid and the CAA grid will be considered. Finally some results of CAA calculations with a Navier-Stokes and Euler mean flow will be presented.

2. The RANS Calculations

As generic example, the viscous flow around a wing section of an airfoil representative for those at modern civil aircraft was calculated. The flow solver was the DLR FLOWer code [1,8,9], which solves the compressible Navier-Stokes equations on a grid consisting of topologically rectangular blocks (structured multi-block grid). The

Paper presented at the RTO AVT Symposium on "Ageing Mechanisms and Control: Part A – Developments in Computational Aero- and Hydro-Acoustics", held in Manchester, UK, 8-11 October 2001, and published in RTO-MP-079(I).

free stream Mach number was $Ma = 0.2$, the Reynolds number $\mathrm{Re} = 1.6 \times 10^6$ and angle of attack of $\alpha = 7°$. As turbulence model, the k-ω model of Wilcox was used.

2.1 The RANS Grid

The RANS grid consists of 12 blocks with a total of about 86000 grid points, cf. Fig.1. The topologically complicated structure of the grid stems from the fact that with the same topology also a wing section with deployed Fowler flap has to be calculated (as part of a future 3d simulation). It can be seen that a typical RANS grid becomes very coarse in the far field region. Any calculation of wave propagation would break down there.

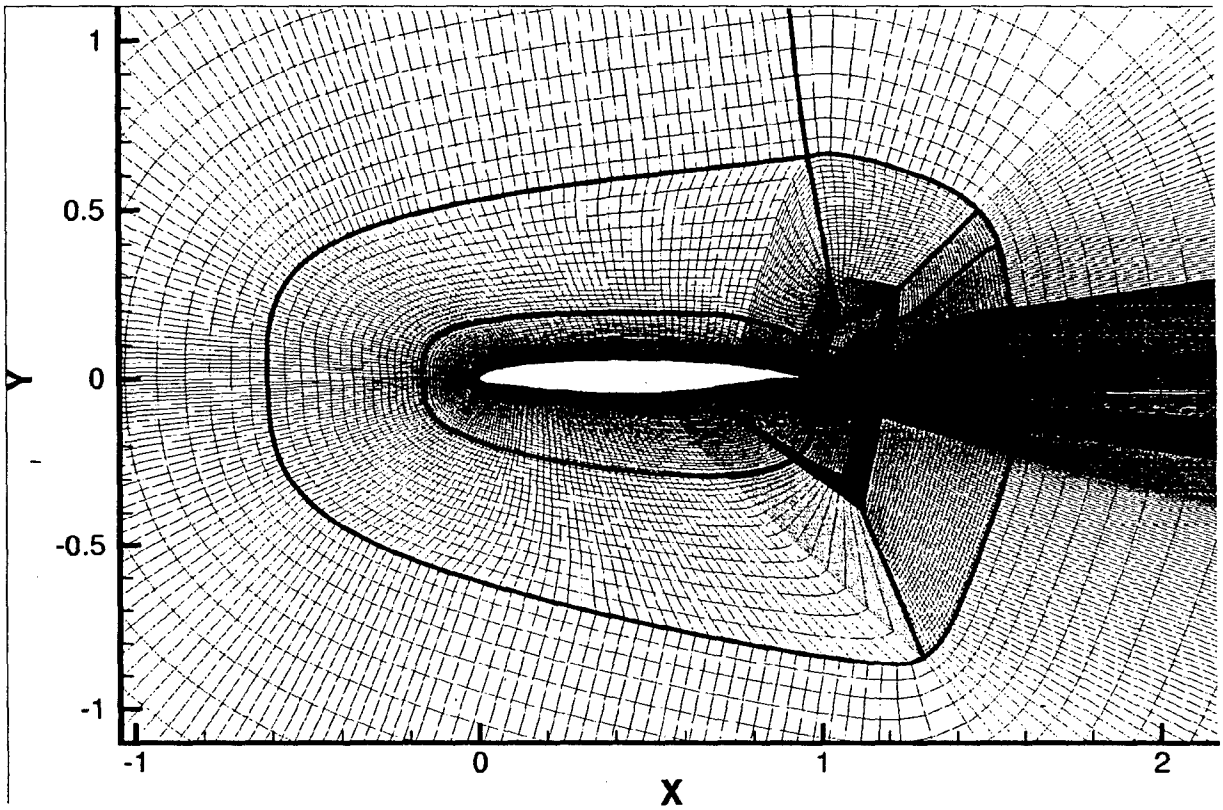

Fig. 1 The RANS grid(detail). (Thick lines are block boundaries)

2.2 The Mean Flow Field

The mean flow field around a wing section at high angle of attack has a strong pressure minimum at the nose of the profile, cf. Fig.3, which corresponds to high local velocities and gradients of the flow variables. The maximum Mach number at the nose is about $Ma = 0.45$ in contrast to a free stream mach number of 0.2. The thin boundary layer requests a small grid spacing there. In the present case the distance of the first grid line from the surface was 1×10^{-5} of the chord length. The examined wing section has a blunt trailing edge with a thickness of about 0.33% of the chord length. This finite trailing edge requests a considerable amount of fine tuning of the grid in order to achieve sufficient convergence of the RANS solution. Besides problems with the turbulence models there, the main reason for this difficulty is the complex grid topology in the wake of the profile. For comparison purposes also an Euler calculation was performed which needed a different grid not shown here.

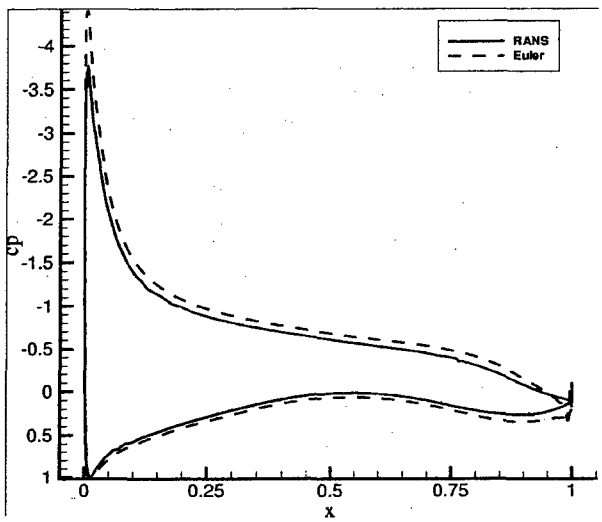

Fig. 2 The pressure distribution of the mean flow

3. Interpolation between RANS- and CAA-Grid

3.1 The CAA Grid

The CAA grid in the present example consists of 2 blocks with a total of about 18000 grid points, cf. Fig.3. The boundary layers of the profile were resolved by approximately 10 grid points with a minimum grid spacing of 5×10^{-6} at the nose of the profile. This results in an allowable time step of the CAA calculation of the same magnitude. The minimum grid spacing at the nose is about the half of the spacing of the RANS grid there. This small spacing was chosen mainly in order to resolve possible small unsteady flow structures. It should be remarked however that the progression of the grid spacing at the nose is larger than the progression choosen in the RANS grid. The gridspacing in the far field was chosen such that one retains enough resolution of the evanescent sound waves and is therefore significantly smaller than in the RANS grid.

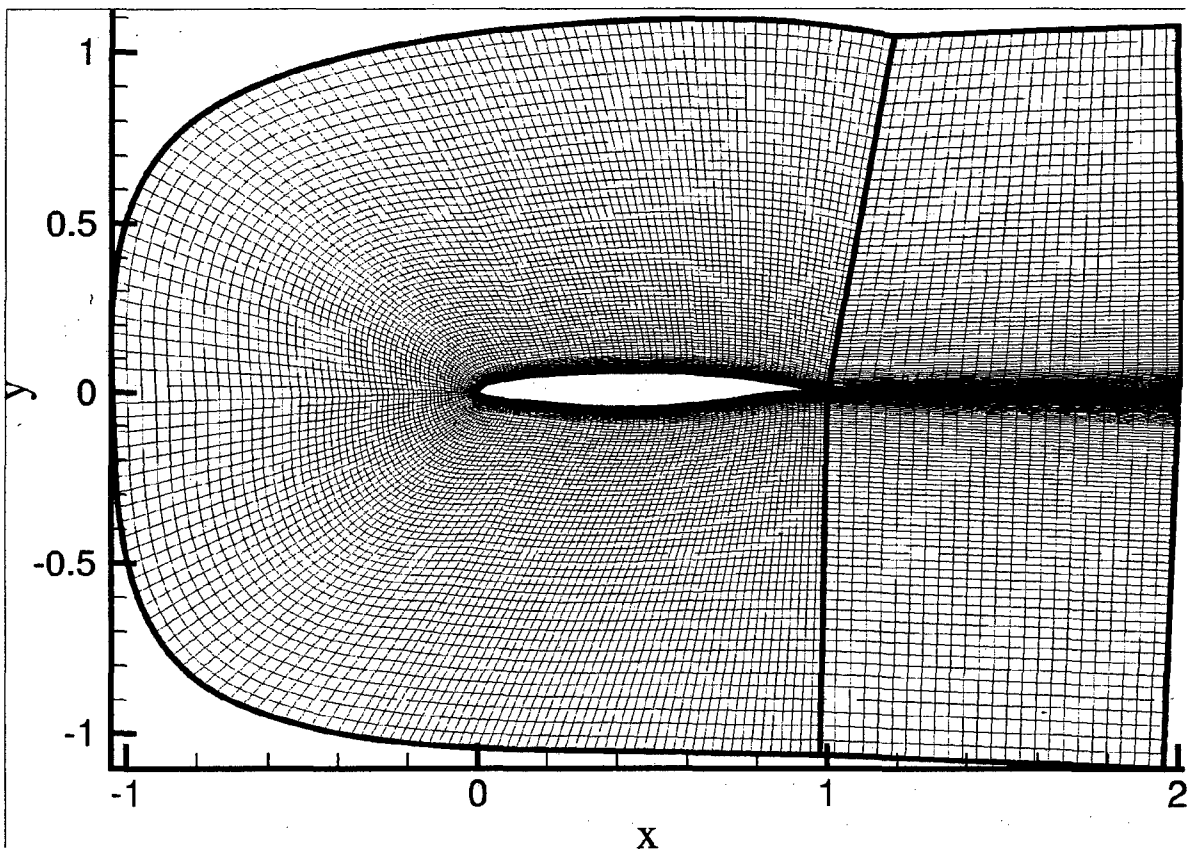

Fig. 3 The CAA grid (2 blocks. Thick lines are block boundaries.)

It should be noted as well, that a considerable coarsening (factor 20) of the CAA-grid normal to the boundary layer close to the surface will still allow for an adequate representation of essential features of the physics. As will be seen later, this is a consequence of the fact that the presence of boundary layers has only a small influence on the scattering problem considered.

3.2 The Interpolation Procedure

The interpolation procedure between the RANS grid and the CAA grid is complicated by two circumstances. First, the two grids can be located almost arbitrary in 3d space (The only restriction is, that the CAA grid has to be enclosed by the RANS grid). This results in an expensive search for appropriate initial conditions for the iterative determination of the parameters of a point of the CAA grid. The second issue is the very fact that in complicated 3d calculations, one has to search some million CAA points in some million RANS cells. This poses great demands upon the speed of the algorithm. Therefore an easy to evaluate, local polynomial approximation was chosen for the representation of the grid functions.

3.2.1 The Interpolating Polynomial

The RANS solver FLOWer developed at DLR utilizes a general curvilinear multi-block structure. Every block of the grid consists of the $n_1 \times n_2 \times n_3$ points

$$\mathbf{x}(i,j,k) = \begin{pmatrix} x_{ijk} \\ y_{ijk} \\ z_{ijk} \end{pmatrix} \quad , \quad 0 < i < n_1 - 1 \quad , \quad 0 < j < n_2 - 1 \quad , \quad 0 < k < n_3 - 1$$

Every flow variable in the FLOWer code is defined at these grid nodes. It is convenient to consider the grid coordinates and every other function defined on the block as function of the grid indices. I.e., we introduce parameters (ξ,η,ζ), in such a way that e.g.

$$\mathbf{x}(i,j,k) = \mathbf{x}(\xi = i, \eta = j, \zeta = k)$$

In the following, the symbol f stands for any variable that is defined at the grid nodes.The interpolation problem can now be formulated as follows

1. Choose an appropriate representation of a function $f(\xi,\eta,\zeta)$ on the block.
2. Determine for a given point $\mathbf{y}$ inside the block the values (ξ,η,ζ) from the vector equation $\mathbf{y} = \mathbf{x}(\xi,\eta,\zeta)$.

Since one has to interpolate a vast amount of points (up to some millions in 3d calculations), one needs a fast interpolation method with acceptable accuracy. This leads to a polynomial interpolation of the function f on every cell of the block. Therefore, we consider now a cell $i \le \xi \le i+1$, $j \le \eta \le j+1, k \le \zeta \le k+1$ of the block and introduce local coordinates $u = \xi - i, v = \eta - j, w = \zeta - k$ in this cell. The interpolation problem now boils down to a polynomial interpolation of a function on a unit cell $0 \le u \le 1, 0 \le v \le 1, 0 \le w \le 1$, where the function values (and perhaps appropriate derivates of the function) are given at the corners of this unit cell. The most simple interpolation is the linear one

$$f(u,v,w) = a_{000} + a_{100}u + a_{010}v + a_{001}w + a_{110}uv + a_{101}uw + a_{011}vw + a_{111}uvw$$

The eight coefficients $a_{000},\ldots,a_{111}$ can be determined by the eight function values $f^{000} = f(0,0,0),\ldots,f^{111} = f(1,1,1)$ at the corners of the cell. Unfortunately the accuracy of this interpolation is quite poor. An improvement however can be achieved using a higher order polynomial. In this case one needs additional information from the neighboring cells or from derivatives at the corners of the cell in order to determine the higher number of coefficients. In order to construct a local approximation the second approach was choosen,

using appropriate values for the 2nd derivatives of f. It was convenient to consider the following polynomial up to third order in u, v and w as interpolating function

$$f(u,v,w) = a_{000} + a_{111}uvw + \begin{pmatrix} a_{110} \\ a_{011} \\ a_{101} \end{pmatrix}^T \begin{pmatrix} uv \\ vw \\ uw \end{pmatrix} + \begin{pmatrix} a_{100} \\ a_{010} \\ a_{001} \end{pmatrix}^T \begin{pmatrix} u \\ v \\ w \end{pmatrix} + \begin{pmatrix} a_{200} \\ a_{020} \\ a_{002} \end{pmatrix}^T \begin{pmatrix} u^2 \\ v^2 \\ w^2 \end{pmatrix} + \begin{pmatrix} a_{210} \\ a_{201} \\ a_{120} \\ a_{021} \\ a_{012} \\ a_{102} \end{pmatrix}^T \begin{pmatrix} u^2v \\ u^2w \\ uv^2 \\ v^2w \\ vw^2 \\ uw^2 \end{pmatrix} +$$

$$+ \begin{pmatrix} a_{211} \\ a_{121} \\ a_{112} \end{pmatrix}^T \begin{pmatrix} u^2vw \\ uv^2w \\ uvw^2 \end{pmatrix} + \begin{pmatrix} a_{300} \\ a_{030} \\ a_{003} \end{pmatrix}^T \begin{pmatrix} u^3 \\ v^3 \\ w^3 \end{pmatrix} + \begin{pmatrix} a_{310} \\ a_{301} \\ a_{130} \\ a_{031} \\ a_{013} \\ a_{103} \end{pmatrix}^T \begin{pmatrix} u^3v \\ u^3w \\ uv^3 \\ v^3w \\ vw^3 \\ uw^3 \end{pmatrix} + \begin{pmatrix} a_{311} \\ a_{131} \\ a_{113} \end{pmatrix}^T \begin{pmatrix} u^3vw \\ uv^3w \\ uvw^3 \end{pmatrix}$$

Now one has to determine the 32 coefficients $a_{000}, a_{100}, \ldots, a_{113}$. Eight equations are provided by the function values at the corners of the cell. Another 24 have to be determined from appropriate derivatives at the corners. Since only the function values are provided by the RANS calculation, one has to determine sufficiently accurate approximations of the derivatives at the grid points numerically. One way to do this is a (one-dimensional) cubic spline interpolation of the function along every grid line [11]. A cubic spline has continuous derivatives up to second order and therefore values for the second derivatives f_{uu}, f_{vv} and f_{ww} at the grid points can easily be obtained. An inspection of the second derivatives of the interpolating polynomial

$$f_{uu} = 2a_{200} + 6a_{300}u + 2a_{210}v + 2a_{201}w + 6a_{310}uv + 6a_{301}uw + 2a_{211}vw + 6a_{311}uvw$$
$$f_{vv} = 2a_{020} + 2a_{120}u + 6a_{030}v + 2a_{021}w + 6a_{130}uv + 2a_{121}uw + 6a_{031}vw + 6a_{131}uvw$$
$$f_{ww} = 2a_{002} + 2a_{102}u + 2a_{021}v + 6a_{003}w + 2a_{112}uv + 6a_{103}uw + 6a_{013}vw + 6a_{113}uvw$$

yields that the coefficients can be calculated successively from the following recursively solvable linear systems (Zero entries are marked with (.))

$$\begin{pmatrix} 2 & . & . & . & . & . & . & . \\ 2 & 6 & . & . & . & . & . & . \\ 2 & . & 2 & . & . & . & . & . \\ 2 & . & . & 2 & . & . & . & . \\ 2 & 6 & 2 & . & 6 & . & . & . \\ 2 & 6 & . & 2 & . & 6 & . & . \\ 2 & . & 2 & 2 & . & . & 2 & . \\ 2 & 6 & 2 & 2 & 6 & 6 & 2 & 6 \end{pmatrix} \begin{pmatrix} a_{200} \\ a_{300} \\ a_{210} \\ a_{201} \\ a_{310} \\ a_{301} \\ a_{211} \\ a_{311} \end{pmatrix} = \begin{pmatrix} f_{uu}^{000} \\ f_{uu}^{100} \\ f_{uu}^{010} \\ f_{uu}^{001} \\ f_{uu}^{110} \\ f_{uu}^{101} \\ f_{uu}^{011} \\ f_{uu}^{111} \end{pmatrix}, \quad \begin{pmatrix} 2 & . & . & . & . & . & . & . \\ 2 & 2 & . & . & . & . & . & . \\ 2 & . & 6 & . & . & . & . & . \\ 2 & . & . & 2 & . & . & . & . \\ 2 & 2 & 6 & . & 6 & . & . & . \\ 2 & 2 & . & 2 & . & 2 & . & . \\ 2 & . & 6 & 2 & . & . & 6 & . \\ 2 & 2 & 6 & 2 & 6 & 2 & 6 & 6 \end{pmatrix} \begin{pmatrix} a_{020} \\ a_{120} \\ a_{030} \\ a_{021} \\ a_{130} \\ a_{121} \\ a_{031} \\ a_{131} \end{pmatrix} = \begin{pmatrix} f_{vv}^{000} \\ f_{vv}^{100} \\ f_{vv}^{010} \\ f_{vv}^{001} \\ f_{vv}^{110} \\ f_{vv}^{101} \\ f_{vv}^{011} \\ f_{vv}^{111} \end{pmatrix}$$

$$\begin{pmatrix} 2 & . & . & . & . & . & . & . \\ 2 & 2 & . & . & . & . & . & . \\ 2 & . & 2 & . & . & . & . & . \\ 2 & . & . & 6 & . & . & . & . \\ 2 & 2 & 2 & . & 2 & . & . & . \\ 2 & 2 & . & 6 & . & 6 & . & . \\ 2 & . & 2 & 6 & . & . & 6 & . \\ 2 & 2 & 2 & 6 & 2 & 6 & 6 & 6 \end{pmatrix} \begin{pmatrix} a_{002} \\ a_{102} \\ a_{012} \\ a_{003} \\ a_{112} \\ a_{103} \\ a_{013} \\ a_{113} \end{pmatrix} = \begin{pmatrix} f_{ww}^{000} \\ f_{ww}^{100} \\ f_{ww}^{010} \\ f_{ww}^{001} \\ f_{ww}^{110} \\ f_{ww}^{101} \\ f_{ww}^{011} \\ f_{ww}^{111} \end{pmatrix}, \quad \begin{pmatrix} 1 & . & . & . & . & . & . & . \\ 1 & 1 & . & . & . & . & . & . \\ 1 & . & 1 & . & . & . & . & . \\ 1 & . & . & 1 & . & . & . & . \\ 1 & 1 & 1 & . & 1 & . & . & . \\ 1 & 1 & . & 1 & . & 1 & . & . \\ 1 & . & 1 & 1 & . & . & 1 & . \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} a_{000} \\ a_{100} \\ a_{010} \\ a_{001} \\ a_{110} \\ a_{101} \\ a_{011} \\ a_{111} \end{pmatrix} = \begin{pmatrix} f^{000} \\ f^{100} - a_{200} - a_{300} \\ f^{010} - a_{020} - a_{030} \\ f^{001} - a_{002} - a_{003} \\ f^{110} - \dots \\ f^{101} - \dots \\ f^{011} - \dots \\ f^{111} - \dots \end{pmatrix}.$$

3.2.2 *Calculation of the Parameters*

Once an interpolating polynomial can be constructed for any function $f(\xi,\eta,\zeta)$, the main problem of the grid to grid interpolation can be tackled, i.e. the determination of the parameter values (ξ,η,ζ) for an arbitrary point $\mathbf{y}$ inside the RANS grid. This can be split into two parts

1. Find the appropriate block of the RANS grid and some initial conditions for the parameter values.
2. Solve the (nonlinear) vector equation $\mathbf{y} = \mathbf{x}(\xi,\eta,\zeta)$ for (ξ,η,ζ) using the polynomial approximation above for every component function of $\mathbf{x}$.

The crucial point in this algorithm is to provide appropriate initial conditions for the parameters. First, $\mathbf{y}$ is checked against the bounding box of a block. If $\mathbf{y}$ is inside the bounding box, initial conditions are searched in every cell of the block. In order to do this, $\mathbf{x}(\xi,\eta,\zeta)$ is approximated on every cell by a linear function. Initial values for (ξ,η,ζ) can than be found by solution of a simple system of linear equations. This procedure works for most of the cells of the RANS grid. If however the faces of a cell have strong curvature and/or if the the aspect ratio of the cells takes very large values this procedure can fail. This can happen especially with cells in the boundary layer at the nose of the profile. Then, a second pass is started which uses the parameter of successfully found neighboring points as initial values. If initial conditions are found, a Newton iteration is performed in order to solve $\mathbf{y} = \mathbf{x}(\xi,\eta,\zeta)$, which uses the full polynomial approximation of $\mathbf{x}$.

3.2.3 *Treatment of Wall Points*

Points at walls can be treated differently from interior ones, since a wall point in the CAA grid has to be one in the RANS grid representation of the wall too. Therefore, the search for initial conditions can be performed only over the surfaces of the RANS blocks. The difficulty which arises is that a wall point lies at a surface of a RANS block and the Newton iteration for the determination of the parameter values may run outside the block which deterioriates the convergence of the procedure. Now, however, one parameter value is known from the wall condition and consequently the search can be performed inside the face of the block (and not in the volume of the wall cell). In this case, the determining vector equation $\mathbf{y} = \mathbf{x}(\xi,\eta,\zeta)$ has to be replaced by a set of two equations, since we are searching two parameter values only. If, for example, $\zeta = const.$ denotes the coordinate of a wall, ξ,η are determined such that the difference vector $\mathbf{y} - \mathbf{x}(\xi,\eta)$ is perpendicular to both surface tangent vectors $\frac{\partial \mathbf{x}(\xi,\eta)}{\partial \xi}$ and $\frac{\partial \mathbf{x}(\xi,\eta)}{\partial \eta}$. This is equivalent to the condition that $|\mathbf{y} - \mathbf{x}(\xi,\eta)|$ is minimum. This approach also solves the problem encountered when small differences between the wall surfaces of both grids exist, originating from slightly different spline approximations in both cases.

3.2.4 *Accuracy of the Interpolation*

In order to check the accuracy of the interpolation, the test function $f(x,y) = x^2 + y^2$ was interpolated from the RANS grid onto the CAA grid. The relative error $(f^{CAA} - f^{RANS})/\max(f^{CAA})$ is below 10^{-4} in regions where strong mean flow gradients are expected, cf. Fig.4. This has proved to be sufficient for the intended purpose. The error displays characteristic extrema where large jumps in gridspacing on the RANS grid occur or the grid lines have some kinks which increase the error in the spline approximations. It should however be noted that far away from the body the RANS solution has almost no gradients anymore and in the vicinity of the body the gridspacing is very small. This also alleviates accuracy problems of the interpolation.

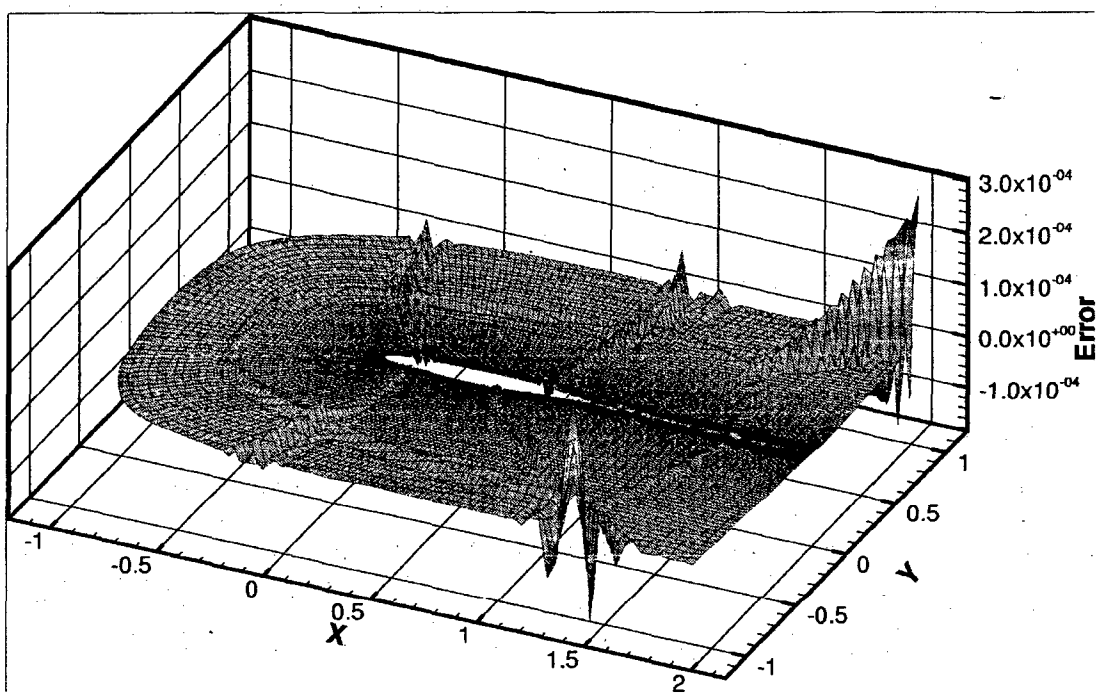

Fig. 4 The relative interpolation error on the CAA grid

4. The CAA Calculations

For the solution of the linearized Euler equations the code PIANO was used, which is currently under development at DLR [5,6]. The LEE solved read in symbolical notation

$$\frac{\partial \rho}{\partial t} + \mathbf{v}_0 \cdot \nabla \rho + \mathbf{v} \cdot \nabla \rho_0 + \rho_0 \nabla \cdot \mathbf{v} + \rho \nabla \cdot \mathbf{v}_0 = 0$$

$$\rho_0 \frac{\partial \mathbf{v}}{\partial t} + \rho_0 \mathbf{v}_0 \cdot \nabla \mathbf{v} + \rho_0 \mathbf{v} \cdot \nabla \mathbf{v}_0 + \rho \mathbf{v}_0 \cdot \nabla \mathbf{v}_0 = -\nabla p$$

$$\frac{\partial p}{\partial t} + \mathbf{v}_0 \cdot \nabla p + \mathbf{v} \cdot \nabla p_0 + \gamma p_0 \nabla \cdot \mathbf{v} + \gamma p \nabla \cdot \mathbf{v}_0 = 0$$

Here ρ and p are the density and pressure disturbance and $\mathbf{v}$ is the vector of the disturbance velocity. The quantities with index 0 are the mean flow values. The equations were brought into dimensionless form with a reference length L, a reference density ρ_∞ and a reference velocity a_∞ which was choosen to be the velocity of sound far away from the body. These equations are solved on a structured multi-block grid, like the one used in the FLOWer code. The spatial derivatives are approximated by the usual 4th order DRP-stencils [14] and time

integration is done by the standard 4th order Runge-Kutta procedure. Inflow and outflow boundary conditions were used according to [14].

4.1 Filtering

Since the utilized DRP-Schemes have no dissipation, an efficient removal of spurious waves, i.e. waves which can not be resolved on the numerical grid, is necessary. The usual way to do this is to use artificial selective damping [13]. This approach is quite expensive, since additional terms have to be calculated in the differential equations. A more convenient way is to apply periodically during the calculation suitable digital filters on the fields [12,15]. In PIANO, symmetric 6th and 8th order filters are implemented. In the present study, the filters were used every hundred time steps. The 6th ($N=3$) and 8th ($N=4$) order filter are defined as [12]

$$\overline{\psi}_i = a_0\psi_i + \sum_{j=1}^{N} a_j(\psi_{i+j} + \psi_{i-j})$$

$$N=3 \;:\; a_0 = 0.6875 \;,\; a_1 = 0.46875 \;,\; a_2 = -0.1875 \;,\; a_3 = 0.03125$$

$$N=4 \;:\; a_0 = 0.7265625 \;,\; a_1 = 0.4375 \;,\; a_2 = -0.21875 \;,\; a_3 = 0.0625$$
$$,\; a_4 = -0.0078125$$

where $\overline{\psi}_i$ denotes the filtered quantity and ψ the unfiltered one. This filter is applied in the computational (ξ,η,ζ)-space in the several space dimensions subsequently. No filtering was used in the first N layers normal to a wall.

4.2 Smoothing Normal to Walls

During the calculations it has been observed that very strong gradients of the flow variables develop in the immediate vicinity of walls, especially at the leading edge of the wing section, which lead to numerical instability. Examination of the calculational results revealed that gridpoint fluctuations of the density normal to the wall are the driving force for those instabilities. Therefore, an ad hoc procedure was applied in order to smooth out those gridpoint fluctuations, which we call 'diagonal smoothing'. If j denotes the direction normal to the wall and $j=0$ the position of the wall, the diagonal smoothing procedure for a flow quantity is defined as

$$\overline{\psi_{i,j}} = a_0\psi_{i,j} + \sum_{k=1}^{N} a_k(\psi_{i-k,j+k} + \psi_{i+k,j+k}) \quad , \quad j = 0,1,\ldots,N-1$$

The coefficients a_j are the same as for the filters above. The value of $N=3$ was used in connection with 6th order filtering and $N=4$ with 8th order filtering. This smoothing procedure was not applied normal to the trailing edge face. In the results presented, all flow quantities were smoothed, although in some calculations it would have been sufficient to smooth the density alone.
It should be emphasized however, that up to now no mathematical justification can be given for this kind of smoothing. This is a problem that clearly requests further investigations in the future.

4.3 Pressure Pulse Hitting the Leading Edge

A common benchmark problem for CAA codes is the scattering of a pressure pulse hitting the leading edge of a wing section. Fig.5 shows the scattering of a pressure pulse located at $x=-0.35, y=0$ at time at $T=0$ in front of the profile for the RANS mean flow and 8th order filtering. It can be seen, that the wave front travels slightly faster above the upper side of the profile due to the higher velocities of the mean flow there. The most remarkable phenomenon, which was not observed in scattering investigations using an Euler mean flow, is however the strong generation of vorticity in the boundary layer at the nose of the profile. This vorticity is convected by the mean flow along the upper surface of the profile. Fig.6 shows the contour plot of the v-velocity component and the local direction of the velocity field by means of momentary streamlines of the disturbance velocity. Fig. 7 shows the corresponding quantities for an Euler mean flow field, where almost no vorticity is generated in the vicinity of the body.

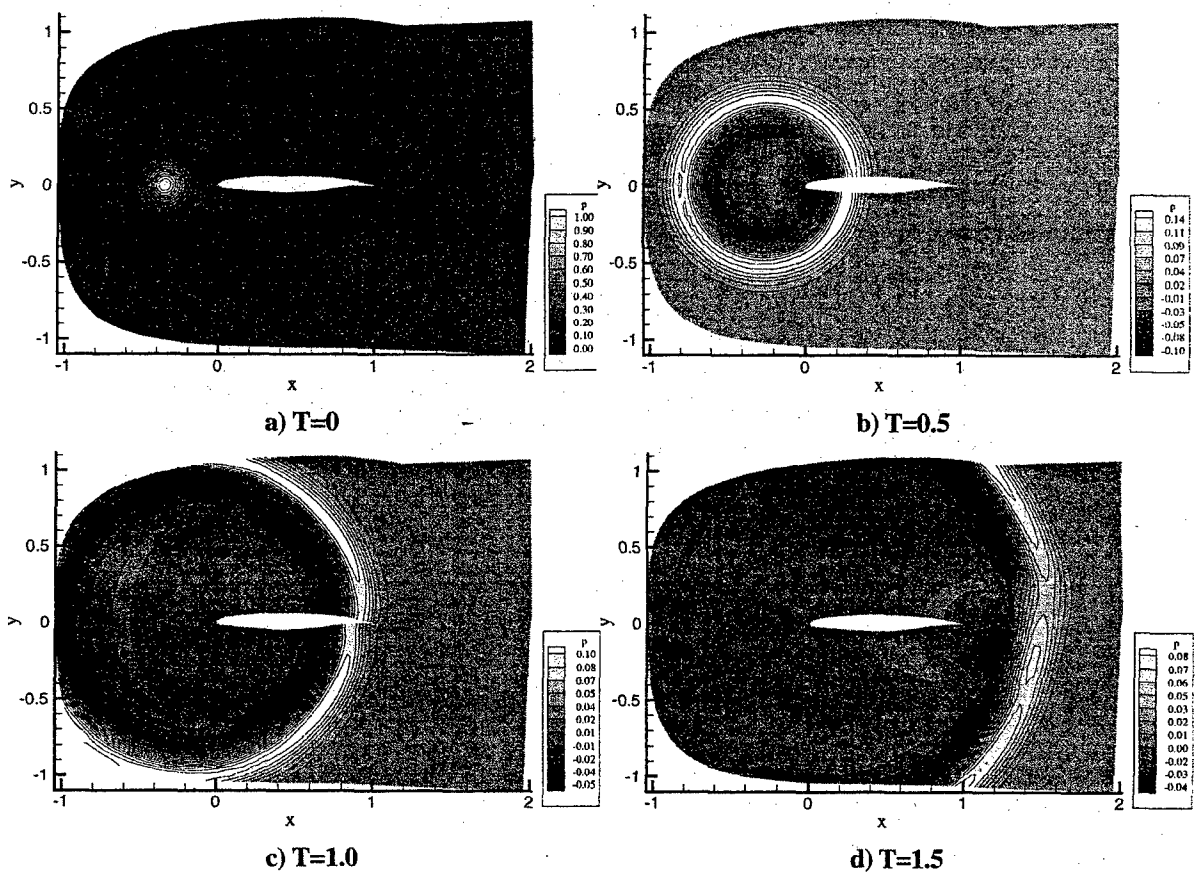

Fig.5 Scattering of a Pressure Pulse (RANS mean flow)

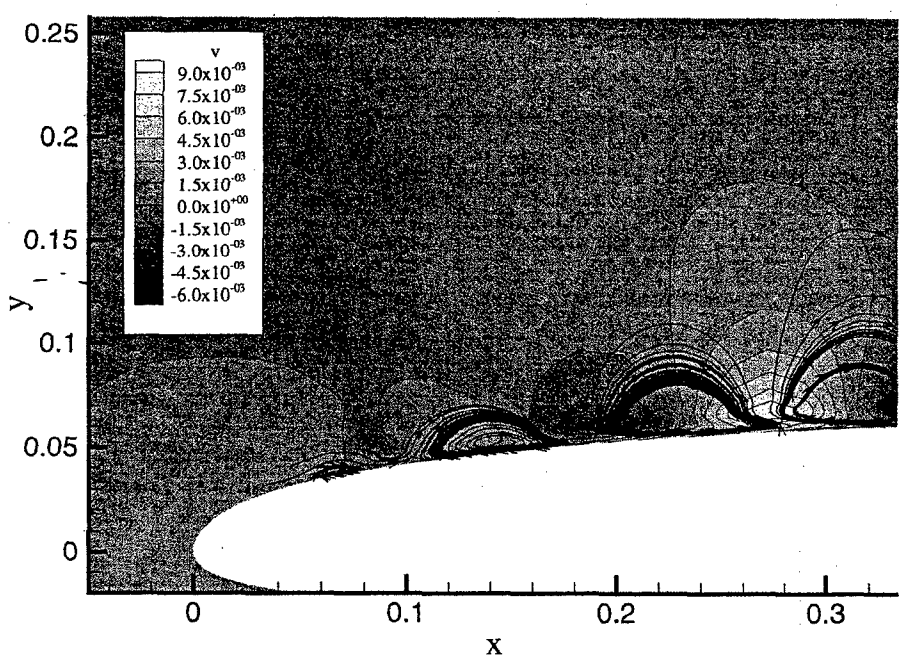

Fig.6 T=1.5, RANS mean flow: v-velocity field with streamlines of the disturbance velocity

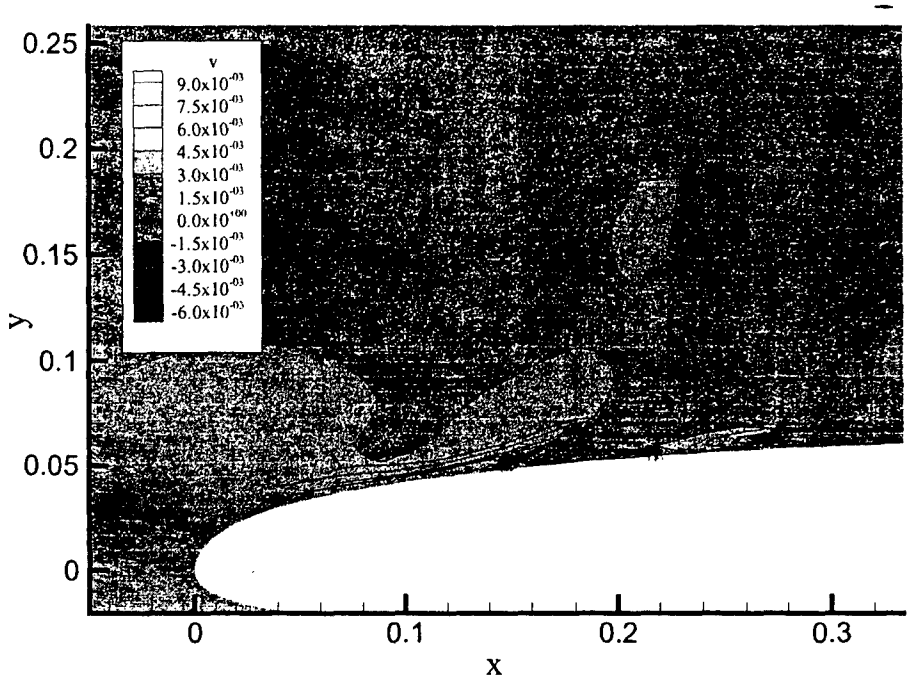

Fig. 7 T=1.5, Euler mean flow: v-velocity field with streamlines of the disturbance velocity

A physical explanation for this strong generation of disturbance vorticity in regions where the mean flow field has vorticity maxima too can be given if one considers the compressible vorticity transport equation:

$$\frac{D\boldsymbol{\omega}}{Dt} = -\boldsymbol{\omega}\nabla\mathbf{v} + (\boldsymbol{\omega}\cdot\nabla)\mathbf{v} + \frac{\nabla\rho\times\nabla p}{\rho^2}$$

After linearization the first term on the right hand side gives a term consisting of the product of the vorticity of the mean flow and the divergence of the disturbance velocity. Since the mean flow vorticity takes huge values in the boundary layer this results in a strong source term for the disturbance vorticity if the pressure wave arrives. Similar vortex structures have been found by Manoha et al. [10] performing a large eddy simulation of the flowfield around a NACA 0012 airfoil.

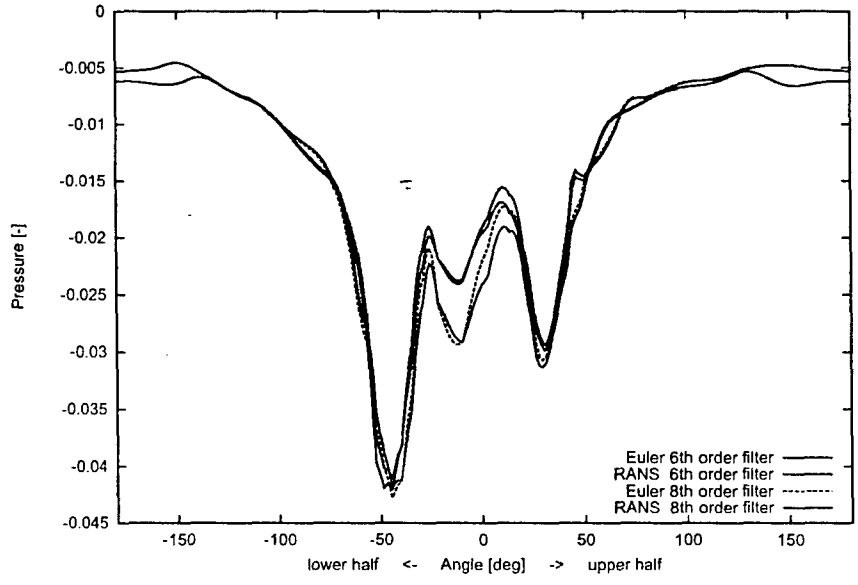

Fig. 8 T=1.5, Pressure on a circle around the profile for different mean flow fields and filters

In order to compare the influence of the different mean flow fields and two different filters, for T=1.5 the pressure on a circle (origin (0.5,0.0), radius 0.8) around the profile was plotted, cf. Fig.8. The angle 0° corresponds to the downstream direction.
It can be seen that the difference between the 6th and the 8th order filters is much larger than the difference between RANS- and Euler mean flow. As one would expect, one recognizes that the pressure profile for the 8th order filter has sharper gradients than that of the 6th order filter. The largest difference between the two filters occurs near the trailing edge of the profile (domain near angle 0°), which becomes clear if on considers that there the sharpest gradients in the disturbance flow field develop. The very small difference between the pressures of the RANS and Euler mean flow fields indicates that almost no sound is generated by the vortex structures mentioned above and that the influence of mean flow field wall boundary layers can be neclected in the scattering problem considered.

5. Conclusions

For the solution of the linearized Euler equations with a RANS mean flow field the RANS grid can normally not be used. Therefore an interpolation procedure for the mean flow field between curvilinear structured multi-block RANS- and CAA-grids was suggested, which is based on a higher order polynomial approximation of the grid variables on each cell. The coefficients of the polynomial are determined from the function values and its 2nd derivatives at the corners of the cell. The 2nd derivatives are determined by one dimensional cubic spline interpolation along the gridlines. Of crucial importance is the provision of suitable initial conditions for the iterative determination of the the parameter values of the grid functions.
The interpolation procedure was applied on the scattering problem of a pressure pulse hitting the leading edge of a wing section at an angle of attack of $\alpha = 7^\circ$. Remarkable is the strong production of vorticity in the nose region of the profile. This vorticity is transported by the mean flow along the upper surface of the profile, but obviously does not produce sound waves, as shows the comparison with calculations using an Euler mean flow. In the scattering problem considered, the differences in the pressure fields of the RANS and Euler mean flow cases are very small. The difference between 6th and 8th order filtering are most pronounced in the wake of the profile, where sharp gradients evolve which were more intensely smoothed by the 6th order filter.
An open problem that remains is the strong mathematical justification of the diagonal smoothing procedure, which had to be applied in immediate vicinity of solid walls, in order to stabilize the calculations.

6. References

[1] Aumann, P.; Barnewitz, H.; Schwarten, H.; Becker, H.; Heinrich, R.; Roll, B.; Galle, M.; Kroll, N.; Gerold, T.; Schwamborn, D.; Franke, M.: "MEGAFLOW: Parallel complete aircraft CFD", Parallel Computing, vol. 27, no. 4, pp. 415-440, 2001.

[2] Bailly, C.; Juvé, D.: "A stochastic approach to compute subsonic noise using linearized Euler's equations", AIAA-paper 99-1872, American Institute of Aeronautics and Astronautics, 1999.

[3] Bailly, C.; Lafon, P.; Candel, S.: "A stochastic approach to compute noise generation and radiation of free turbulent flows", in 1st AIAA/CEAS Aeroacoustics Conference, 1995.

[4] Bailly,C.; Lafon, P.; Candel, S.: "Computation of noise generation and propagation for free and confined turbulent flows", in 2nd AIAA/CEAS Aeroacoustics Conference, 1996.

[5] Grogger, H.A.; Delfs, J.W.; Lauke, T.G.; Lummer, M.; Yin, J.: "Simulation of leading edge noise of airfoils using CAA based on body fitted grids", presented at the International Congress of Acoustics and Vibration ICAV7, Garmisch, Germany, submitted to International Journal of Acoustics and Vibration

[6] Grogger, H.A.; Lummer, M.; Lauke,Th.: "Simulation of the Interaction of a Three Dimensional Vortex with airfoils using CAA", AIAA-Paper No. 2001-2137, (2001)

[7] Kalitzin, N.; Wilde, A.: "Application of the stochastic noise generation and radiation model to trailing edge noise", in Aeroacoustic Workshop in connection with the German research project SWING - Proceedings (P. Költzsch and N. Kalitzin, eds.), (Dresden), TU Dresden, 1999

[8] Kroll, N.; Rossow, C.-C.; Becker, K.; Thiele, F.: "MEGAFLOW - a numerical flow simulation system", ICAS-Paper 98-2.7.4, 1998.

[9] Kroll, N.; Rossow, C.-C.; Becker, K.; Thiele, F.: "The MEGAFLOW project", Aerospace Science and Technology, vol. 4, no. 4, pp. 223-237, 2000.

[10] Manoha, E.; Delahay, C.; Redonnet, S.; Ben Khelil, S.; Guillen, P.; Sagaut, P.; Mary, I.: "Numerical prediction of the unsteady flow and radiated noise from a 3D lifting airfoil", NATO RTO-AVT Symposium on Aging mechanisms and Control, Part A – Development in Computational Aero- and Hydro-Acoustics, Manchester, UK, 8-11 Oct. 2001.

[11] Press, W.H.; Flannery, B.P.; Teukolksky, S.A.; Vetterling, W.T.: "Numerical Recipes", Cambridge: Cambridge University Press, 1986.

[12] Shang, J.S.: "High-order compact-difference schemes for time-dependent maxwell equations", Journal of Computational Physics, vol. 153, pp. 312-333, 1999.

[13] Tam, C.K.W.; Webb, J.C.; Dong, Z.: "A study of the short wave components in computational acoustics", Journal of Computational Acoustics, vol. 1, pp. 1-30, (1993)

[14] Tam, C.K.W.; Webb, J.C.; "Dispersion-Relation-Preserving Finite Difference Schemes for Computational Aeroacoustics", Journal of Computational Physics, vol. 107, pp. 262-281, (1993)

[15] Vasilyev, O.V.; Lund, T.S.; Moin, P.: "A general class of commutative Filters for LES in complex geometries", Journal of Computational Physics, vol. 146, pp. 82-104, 1998.

Reference # of Paper: 16
Discusser's Name: Prof. J. J. McGuirk
Author's Name: Dr. Markus Lummer
Question:

The RANS solution is converged in the sense that mass and momentum balances on the RAND cells. Do you check that after interpolation the interpolated mean flow satisfies mass and momentum balance on the CAA grid? If not, the linearized Euler equations you solve in a discretized sense may contain artificial diffusive fluxes other than zero on their right hand sides.

Answer:

The balance of the interpolated fluxes is not checked. Since the linearized Euler equations are solved in nonconservative form, artificial fluxes are always present on the right hand side: but are assumed to be small. We assume that the modeling error in the RANS calculations is at least one order of magnitude larger than possible errors introduced by the interpolation.